WebRTC Scale-Up — пред-релизный чек-лист

30 пунктов перед масштабированием WebRTC-продукта — SFU, LastN, каскад, мост, AI-агенты

1. Базовый один SFU

- [ ] Потолок участников на узел замерен под реальной нагрузкой — не по заявлениям вендора.
- [ ] Egress NIC и CPU мониторятся и алертятся при 70% утилизации.
- [ ] DTLS handshake failure rate алертится с причиной (cert mismatch, версия, cipher).
- [ ] Супервизия процесса: SFU чисто рестартится, звонки дренируются при shutdown.
- [ ] Нагрузка RTCP feedback и ICE consent freshness замерена на пике.

2. LastN, simulcast, SVC

- [ ] LastN-форвардинг включён по умолчанию для конференций больше ~12 участников.
- [ ] Simulcast-слои (обычно 3) публикуются каждым видеосендером — проверить в chrome://webrtc-internals.
- [ ] Переключение доминирующего говорящего стабильно — нет быстрого мерцания при crosstalk.
- [ ] RTP-расширение audio-level (RFC 6464) согласовано на каждом публикаторе.
- [ ] SVC (VP9 или AV1) рассмотрен для следующего поколения; H.264 simulcast приемлем в 2026.

3. Дизайн каскада

- [ ] Регионы выбраны по плотности пользователей: us-east-1, eu-central-1, ap-northeast-1, ap-southeast-1.
- [ ] Маршрутизация «комната → узел»: политика locality vs room affinity задокументирована.
- [ ] Межузловой транспорт работает по приватному бэкбону, не по публичному интернету.
- [ ] Межузловая аутентификация: mutual TLS или HMAC pre-shared key, ротация задокументирована.
- [ ] Drill отказа меша: один узел убит в staging; остальные перехватывают за 10 секунд.

4. Слой координации

- [ ] Кворумный реестр (Redis cluster, etcd, Hazelcast), а не один Redis-инстанс.
- [ ] Распространение состава комнаты: sub-second (50–200 мс типично).
- [ ] Drill сбоя координации: кластер теряет узел, состояние комнаты выживает.
- [ ] Реестр размерен под тот же blast radius, что и флот SFU — те же SLA и ops-бюджет.
- [ ] Размер состояния на комнату ограничен; большие комнаты шардируют реестр по ключам.

5. Broadcast-мост (аудитория > 5K)

- [ ] LL-HLS / chunked CMAF-упаковщик входит в SFU как обычный участник.
- [ ] Раскладка активного говорящего протестирована — чистые склейки, без мерцания.
- [ ] CDN: минимум один продакшн-CDN (Cloudflare Stream, Fastly, CloudFront, Akamai).
- [ ] Бюджет латентности broadcast задокументирован: 2–5 секунд glass-to-glass приемлемо.
- [ ] Путь реакций / чата: WebSocket-loop обратно в интерактивный тиер.

6. Флот AI-агент-worker-ов

- [ ] Фреймворк выбран: LiveKit Agents 1.5+ или Pipecat 1.0+ (или коммерческий аналог).
- [ ] Worker агента co-located с региональным SFU — та же AZ, в идеале тот же VPC.
- [ ] «Конец речи → старт ответа» измерен p50/p95; цель <800 мс p50.
- [ ] Конвейер VAD → STT → LLM → TTS инструментирован постадийно; bottleneck идентифицирован.
- [ ] Обработка прерываний протестирована: TTS останавливается, частичный transcript коммитится.